Practical - 5

2CS701 – Compiler Construction

Harshit Gajipara

19bce059

**Aim:**

To implement a calculator in YACC: Syntax Directed Translation Use YACC to Write a Grammar for multiple expression statements, and apply syntax directed translation for calculator.

**Code:**

**Calculator.l**

%{

#include <stdio.h>

#include "y.tab.h"

extern int yylval;

%}

%%

[0-9]+ {yylval = atoi(yytext);

return NUMBER;}

[\t] ;

[\n] return 0;

. return yytext[0];

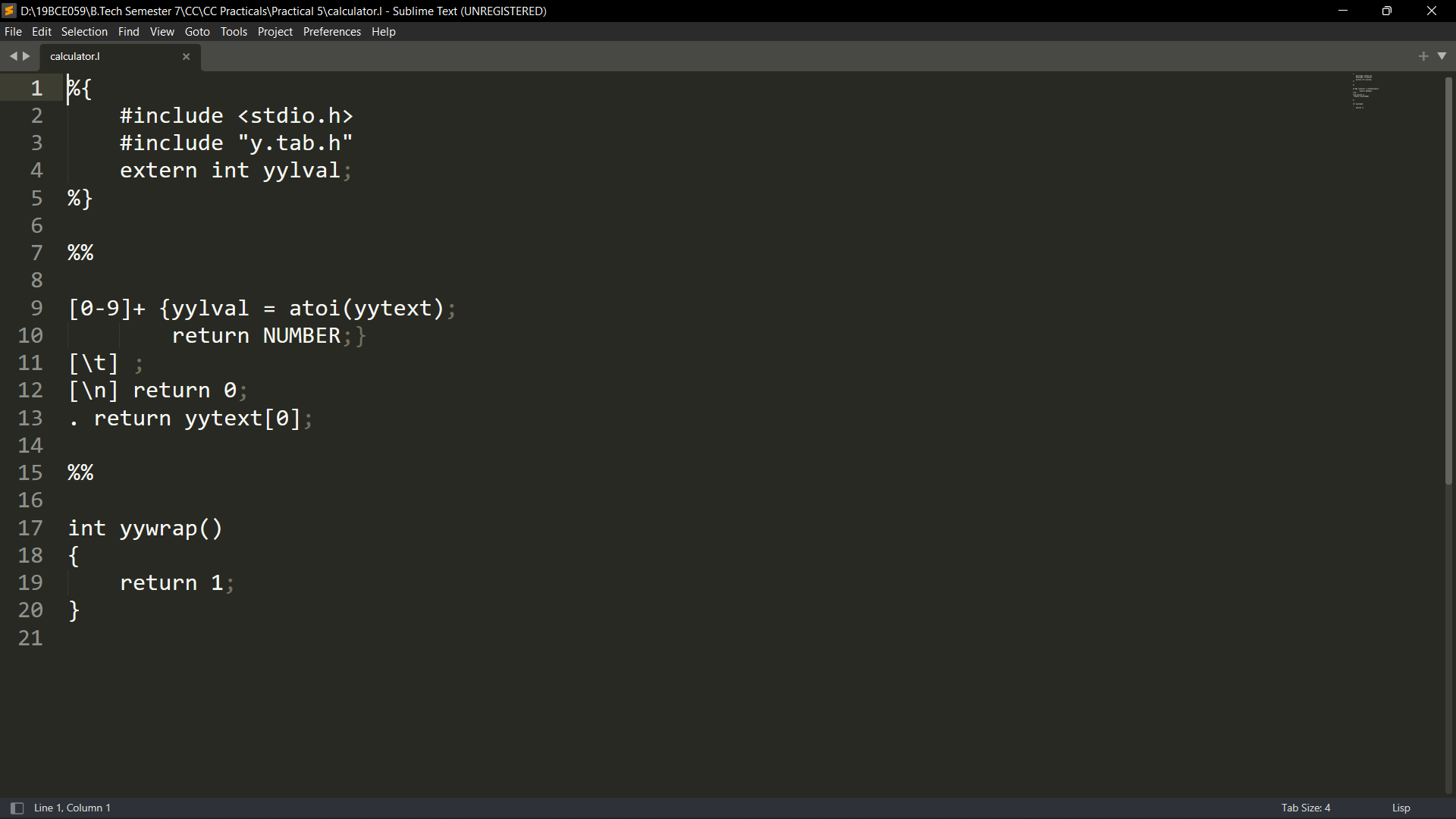
%%

int yywrap()

{

return 1;

}



**Calculator.y**

%{

#include <stdio.h>

int flag=0;

%}

%token NUMBER

%left '+' '-'

%left '\*' '/' '%'

%left '(' ')'

%%

Expression : E{

printf("\nResult = %d\n",$$);

return 0;

}

E : E'+'E {$$=$1+$3;}

| E'-'E {$$=$1-$3;}

| E'\*'E {$$=$1\*$3;}

| E'/'E {$$=$1/$3;}

| E'%'E {$$=$1%$3;}

| '-'E {$$=-$2;}

| '('E')' {$$=$2;}

| NUMBER {$$=$1;}

;

%%

void main()

{

while(1)

{

printf("\nEnter Arithmatic Expression : ");

yyparse();

if(flag==0)

printf("Valid Expression!\n");

}

}

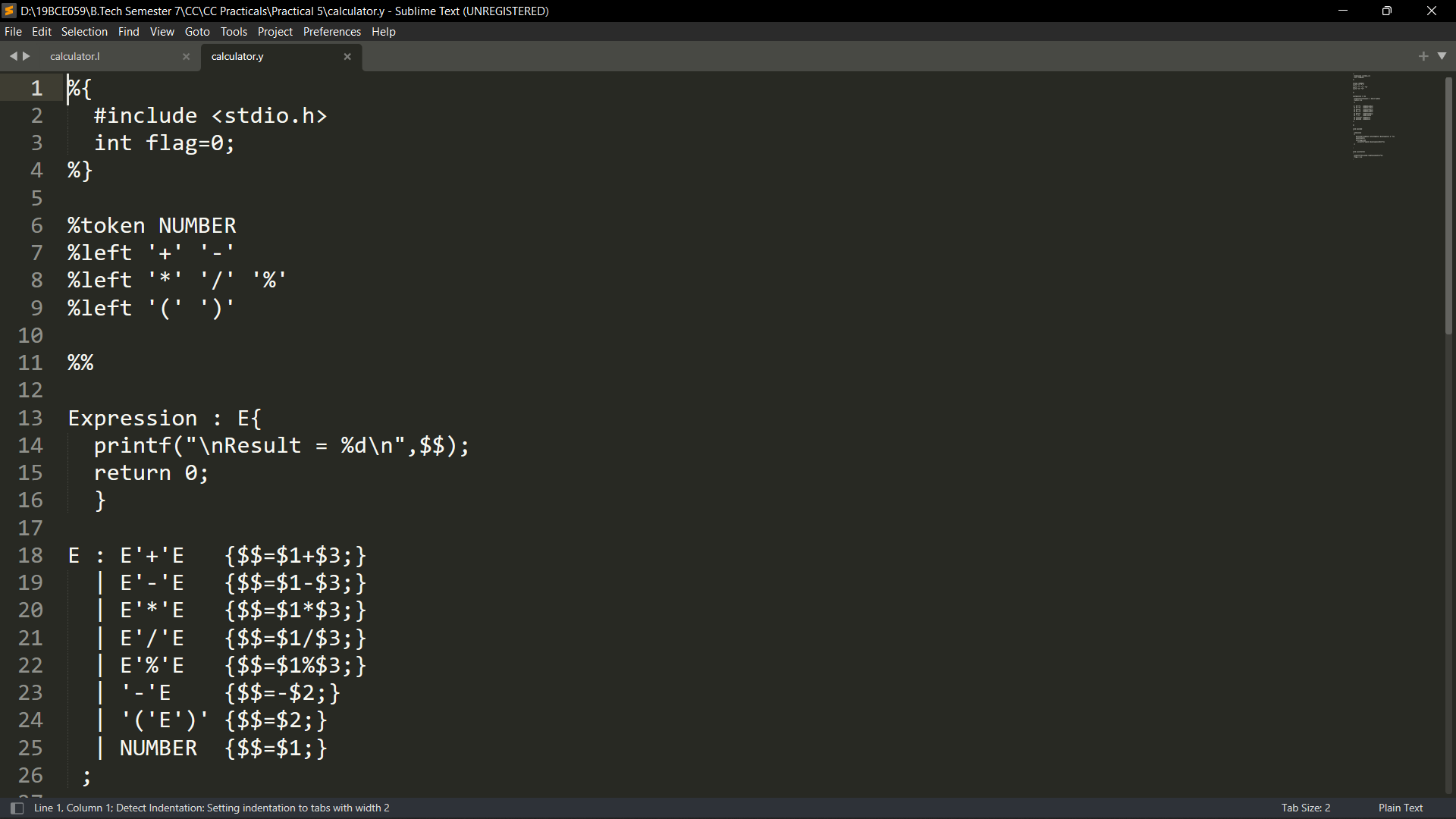
void yyerror()

{

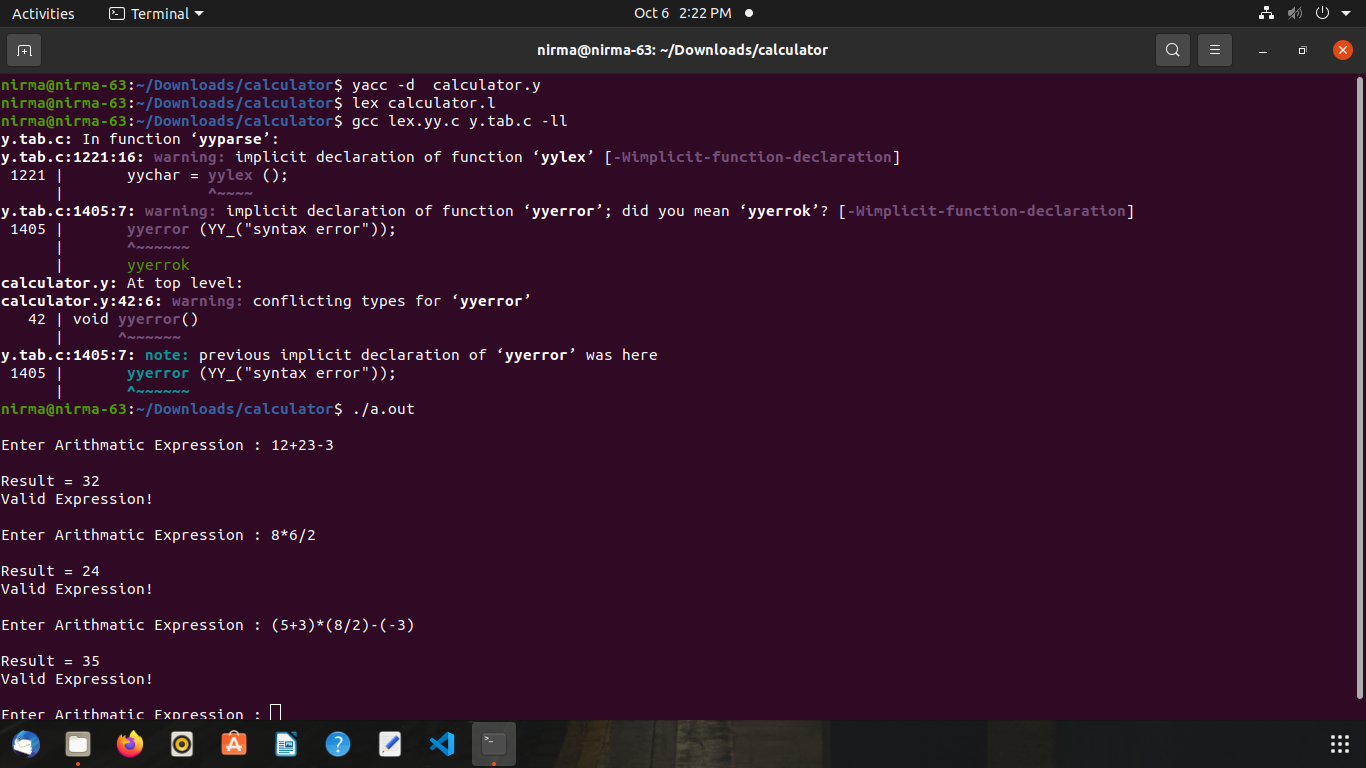
printf("Invalid Expression!\n");

flag = 1;

}



**Output :**

****

**Conclusion:**

In this practical, we learnt how we can use yacc (yet another compiler compiler) to make a syntax directed translation program with help of lex. We can also make any type of grammer and use yacc to parse it.